C

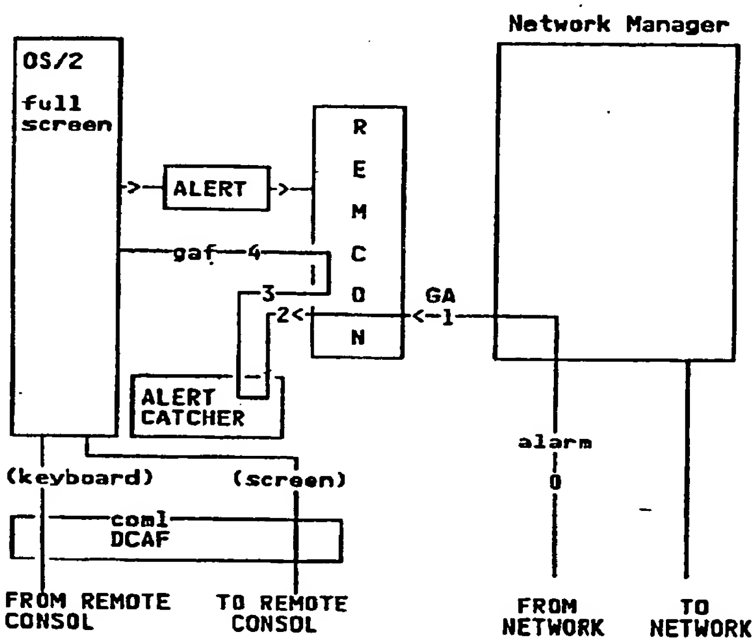
IBM Technical Disclosure Bulletin Vol. 36 No. 08 August 1993

G06F11/22R

Remote Console Session

S06F201:331
S06F201:304
S06F201:018
S06F201:307

p.93-97

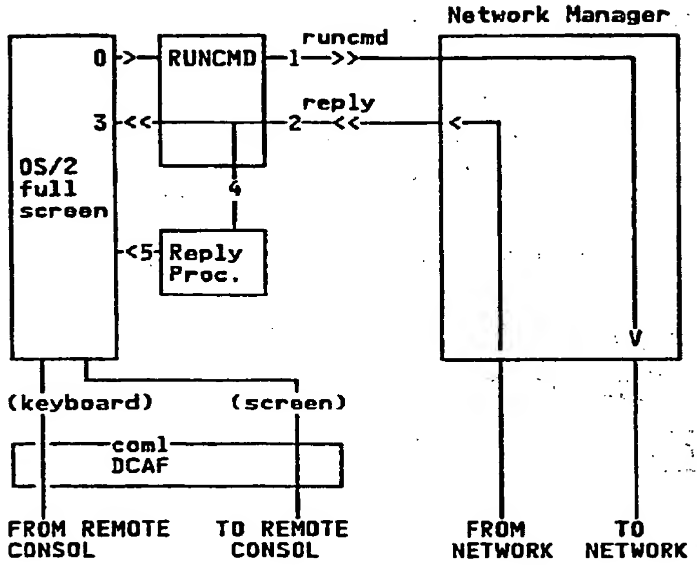


0 : Event occurs in the network and alarm is sent to Network manager
1 : alarm is formated into alert and sent to REMCON
2 : REMCON sends the NMVT to alert cacher for interpretation
3 : Alert catcher sends back to REMCON interpreted alert
4 : REMCON display the alert on the OS/2 full screen

Fig. 1

Remote maintenance of complex networks and possibility to automatically respond to network events is of prime importance as networks need to operate almost unattended and can be distributed covering countries.

The present invention relates to means providing capability of remote maintenance and automation using plain OS/2* session in which a remote console session can be used for remote maintenance thru Public Switched Telephone Network (PSTN) and can used to create automats.



```
0 : Operator issue a command
1 : Command is sent to Network Manager which execute it
2 : Network Manager sends back a reply
3 : Reply is displayed on OS/2 full screen
4 : OR reply triggers execution of a user program
5 : The user program display on the OS/2 full screen
```

Fig. 2

The remote console session uses a standard OS/2 Fullscreen Session (FS) or Window Session (WS) and in the background the remote console session application is started with different asynchroneous OS/2 threads.

The Remote Console application communicates with a Network Manager application which has access to all the nodes in the network. The Remote Console application can be in the same hardware unit as the Network Manager (a PC or a PS/2).

The Remote Console application monitors unattended alarms communi cated by the Network Manager application which are transformed into Generic Alerts and accepts user commands or run commands (RUNCMDs) to be sent to the Network Manager application.

The Remote Console session can be used in two ways:

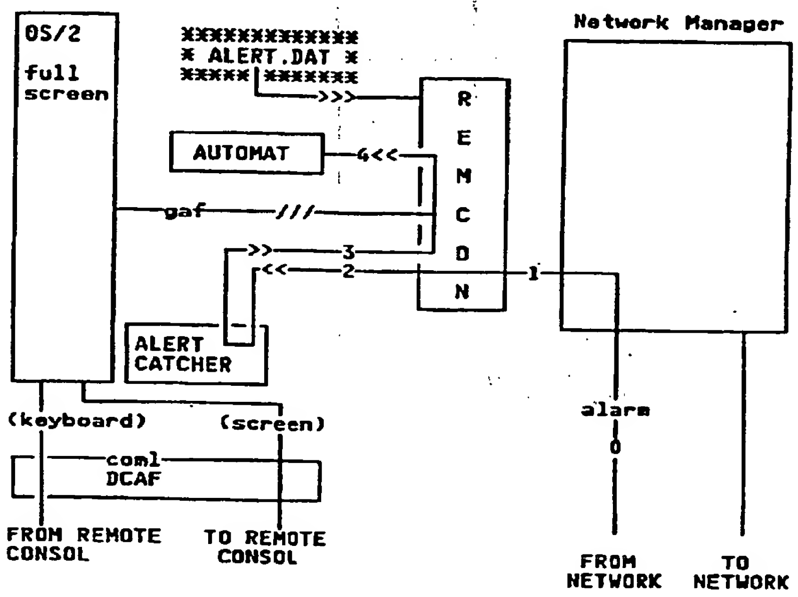
1. Locally at the Network Manager level using the Remote Console session user interface,

At a remote terminal connected via a Public Switched Telephone Network (PSTN) to the Network Manager node with IBM Distributed Control Access Facility (DCAF) installed to allow the remote operator to monitor the Remote Console session using the display and keyboard as he was in front of the local terminal.

The Remote Console process includes two threads:

a. The main thread which is started by invoking REMCON.EXE applica tion either automatically using STARTUP.CMD or manually by opening and OS/2 window and invoking REMCON.EXE.

b. The child thread which is used to open a secondary session in the same OS/2 session.



0 : Event occurs in the network and alarm is sent to Network manager
1 : alarm is formatted into alert and sent to REMCON
2 : REMCON sends the NMVT to alert cacher for interpretation
3 : Alert catcher sends back to REMCON interpreted alert
4 : REMCON analyse the alert, and, if match is found with recorded criteria, it starts an AUTOMAT program to take automatic operator action

Fig. 3

As a result, REMCON.EXE operates in the background and receives the alarms from the Network Manager application and in the foreground the child thread monitors in an OS/2 window the operator activity by provid ing a standard OS/2 session interface to the operator.

The operator session can be controlled remotely using DCAF to moni tor the Remote Console session thru, for instance, an asynchroneous communication port connected thru a PSTN.

The alarms forwarded to REMCON.EXE are formatted into Generic Alerts using a local alert catcher and are used to trigger user's com mand list (CLIST) which can be written in OS/2 REXX or other languages.

Fig. 1 shows the flow for receiving a generic alert:

The CLISTs can include RUNCMDs allowing thus to automate a set of actions following a given event, time of day, event in the network, etc.

Fig. 2 shows the flow for executing a run command:

The following scenario is an example of event/action which can be applied:

- when an alarm from the Network Manager application is passed to REMCON, it is sent to the alert catcher application in the REMCON ses sion for formatting into a NetView Generic Alert. The alerts can be formatted into different formats under option:
  a. a short format limited to an alert ID, a description, a probable cause and a message,
  b. a long format including in addition to a), date/time, hierarchy, different causes and product identifiecation,
  c. a NMVT format which represents the information in a binary form compatible with NetView (hexadecimal values).
- when the formatted alert is received from the alert catcher ap plication, it is analyzed and compared to a set of user's filters pre pared in the ALERT.DAT file in which alert identifier (ALID), alert severity level, alert key data fields have been entered in plain text by the operator,
- if a match is found on the three first fields the alert is dis played, if a match is also found on the data field and if the command field contains an executable program (automated operator), then the program is started via an automated operator with as parameters the differents fields. The automated operator is started in a different OS/2 session to allow several programs (.CMD or .EXE) to execute concurrently.

A typical execution statement in ALERT.DAT will be:

```
ALARM=225,DIS=Y,SEV=*,KEYS=*,CMD=.\SAMPLE.CMD
```

where ALARM=225 is the alarm number 225 which, when it occurs, will trigger automatically the execution of SAMPLE.CMD in current directory.

Fig. 3 shows the flow for executing an automated command:

SAMPLE.CMD can contains any OS/2 statements (or other executables) and also RUNCMDs that REMCON.EXE will pass to the Network Manager appli cation for execution. Execution which can result in other alarms which, if filtered in ALERT.DAT, can trigger other actions and so forth.

The RUNCMDs may have replies which are sent back to the session that started the RUNCMD. The RUNCMD reply can be displayed into two formats, a readable character string or binary NMVT. If the RUNCMD does not execute properly or timeouts, a sense code is returned for analysis by the program which initiated the RUNCMD.

If the RUNCMD is executed successfully, REMCON.EXE looks for an executable (.CMD or .EXE) with a filename derived from the RUNCMD and if it is found, it is executed with the following set of parameters passed:

1. Type of the reply: F if formatted, S if sense code,
2. Number of parameters used with the RUNCMD,
3. Commands and paramaters of used in the RUNCMD,
4. Reply returned by the RUNCMD.

If no executable is found, the reply is directly displayed on the screen as it is received.

* Trademark of IBM Corp.